

BUNDESREPUBLIK DEUTSCHLAND

PRIORITY DOCUMENT

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)



PCT EP03/08081

REC'D 14 NOV 2003

WIPO

PCT

Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

Aktenzeichen: 102 40 000.8

Anmeldetag: 27. August 2002

Anmelder/Inhaber: PACT XPP Technologies AG, München/DE

Bezeichnung: Bussysteme und Rekonfigurationsverfahren

IPC: G 06 F 13/40

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

München, den 2. September 2003
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

CERTIFIED COPY OF
PRIORITY DOCUMENT



Akte: PACT16a

Anmelder: Pact XPP Technologies AG
Muthmannstr. 1
80939 München

5 Vertreter: Patentanwalt
Claus Peter Pietruk
Heinrich-Lilienfein-Weg 5
D-76229 Karlsruhe
Vertreter-Nr. 321 605

10

Deutsche Patentanmeldung

Titel: Bussysteme und Rekonfigurationsverfahren

15 Die vorliegende Erfindung befaßt sich mit Verfahren und Ausgestaltungen von Bussystemen für konfigurierbare Architekturen. Die Optimierung der Konfigurations- und Rekonfigurationsineffizienz wird besonders berücksichtigt.

20 Aufgabe der Erfindung ist es, Neues für die gewerbliche Nutzung bereitzustellen.

Die Lösung der Aufgabe wird unabhängig beansprucht. Bevorzugte Ausführungsformen befinden sich in den Unteransprüchen.

25

Unter einer rekonfigurierbaren Architektur, werden Bausteine (VPU) mit konfigurierbarer Funktion und/oder Vernetzung verstanden, insbesondere integrierte Bausteine mit einer Mehrzahl von ein- oder mehrdimensional angeordneten arithmetischen und/oder logischen und/oder analogen und/oder speichernden und/oder vernetzenden Baugruppen (im Folgenden PAEs) genannt) und/oder kommunikativen/peripheren Baugruppen (IO), die direkt oder durch ein oder mehrere Bussystem(e) miteinander

1

Akte: PACT16a

der verbunden sind. PAEs sind beliebiger Ausgestaltung, Mischung und Hierarchie angeordnet. Diese Anordnung wird im Weiteren PAE-Array oder PA genannt.

Zur Gattung dieser Bausteine zählen systolische Arrays, neuronale Netze, Mehrprozessor Systeme, Prozessoren mit mehreren Rechenwerken und/oder logischen Zellen, Vernetzungs- und Netzwerkbausteine wie z.B. Crossbar-Schalter, ebenso wie bekannte Bausteine der Gattung FPGA, DPGA, XPUTER, etc.. Hingewiesen wird insbesondere in diesem Zusammenhang auf die folgenden Schutzrechte desselben Anmelders: P 44 16 881.0-53, DE 197 81 412.3, DE 197 81 483.2, DE 196 54 846.2-53, DE 196 54 593.5-53, DE 197 04 044.6-53, DE 198 80 129.7, DE 198 61 088.2-53, DE 199 80 312.9, PCT/DE 00/01869, DE 100 36 627.9-33, DE 100 28 397.7, DE 101 10 530.4, DE 101 11 014.6, PCT/EP 00/10516, EP 01 102 674.7, PACT19, PACT20, PACT21, PACT27, PACT28, PACT15. Diese sind hiermit zu Offenbarungszwecken vollumfänglich eingegliedert.

20 1. Aufbau von Bussystemen

Eine herkömmliche Implementierung der Konfiguration erfordert eine Synchronisation zwischen den Objekten. Diese Synchronisation wird zentral über einen FILMO (vgl. PACT04, PACT05, PACT10, PACT17) realisiert. Dadurch vergehen zwischen dem Ende einer alten Konfiguration (reconfig-Event) und dem Beginn einer neuen Konfiguration (Object geht erneut in den Zustand "configured") mindestens so viele Takte wie der gepipelinete CM-Testbus lang ist (Hin- und Rückweg).

30 Zur Beschleunigung werden erfindungsgemäß zwei Verfahren vorgeschlagen:

a) Die erforderliche Reihenfolge wird durch zusätzliche Logik in den Objekten sichergestellt, z.B. Verwaltung von IDs,

Akte: PACT16a

b) Die Objekte werden derart modifiziert, dass keine Reihenfolge mehr beachtet werden muß, sondern daß die ordentliche Verschaltung durch die Architektur der Objekte sichergestellt wird.

5

Für die folgenden Betrachtungen werden die in einer typischen rekonfigurierbaren Architektur vorhandenen Module in zwei Gruppen eingeteilt:

Busse Diese Gruppe umfaßt die Verbindungsleitung zwischen
10 zwei Segmenten. Sie wird repräsentiert durch den Segment-Switch an seinem Ende.

Objekt In dieser Gruppe werden alle Objekte zusammengefaßt, die einen Anschluß an einen Bus, bzw. mit ihrer Umwelt kommunizieren besitzen, d.h. beliebige PAE (z.B. Speicher, ALUs),
15 IO, etc.

Abhängigkeiten bestehen typischerweise zwischen allen direkt benachbarten Objekten. Dies sind im einzelnen:

Bus zu Bus Ein Bus wird dabei repräsentiert durch den Segment-Switch am Ende eines Busses
20

Objekt zu Bus Objekt ist dabei beliebig aus FREG, BREG, ALU und RAM zu wählen. Ebenso zählt alles als Objekt in diesem Sinne, was einen Connect besitzt.

Objekt zu Objekt Diese sind nicht direkt benachbart, es liegt
25 immer ein Bus dazwischen. Hier besteht keine Abhängigkeit.

1.1 Bus zu Bus Abhängigkeit

Längere Busse werden nach dem Stand der Technik beispielsweise von hinten nach vorne konfiguriert. Dabei wird das letzte
30 Bus-Segment mit offenem Switch konfiguriert, alle anderen mit geschlossenem Bus-Switch. Die Einhaltung der Reihenfolge ist erforderlich, um zu verhindern, dass Daten von einem weiter

3

Akte: PACT16a

vorne liegenden Bus in einen weiter hinten liegenden laufen, der noch zu einer anderen Konfiguration gehört.

1.2 Bus zu Objekt Abhängigkeit

- 5 Nach dem Stand der Technik darf ein Objekt erst dann konfiguriert werden, wenn sichergestellt ist, dass die von dem Objekt benutzten Busse bereits konfiguriert sind. Auch diese Abhängigkeit besteht um sicherzustellen, dass keine Daten in eine fremde Konfiguration laufen (PAE-Ausgang) bzw. von einer fremden Konfiguration genommen werden (PAE-Eingang).
- 10

- Zusammenfassend lässt sich feststellen, dass eine Abhängigkeit immer dann besteht, wenn ein Objekt eine Verbindung zu einem anderen Objekt herstellt (Connection-Mask bzw. geschlossener Switch). Dabei darf die Verbindung erst dann hergestellt werden, wenn sichergestellt ist, dass das Objekt, zu dem die Verbindung erstellt werden soll, bereits zur gleichen Konfiguration gehört (d.h. bereits entsprechend konfiguriert ist).
- 15
- 20

2. Kontrolle über ID-Verwaltung

- Der einfachste Ansatz ist, in jedem Objekt die ID oder Array-ID (vgl. PACT10) zu speichern, die das Objekt aktuell verwendet. Sobald eine Verbindung zwischen zwei Objekten konfiguriert wird (z.B. zwischen einem PAE-Ausgang und einem Bus), ist im Voraus zu prüfen, ob beide Objekte dieselbe ID/Array-ID besitzen. Ist dies nicht der Fall, darf die Verbindung nicht hergestellt werden.
- 25
- 30

Obwohl dieses Verfahren grundsätzlich vergleichsweise trivial ist, erfordert es einen hohen Hardware-Aufwand, da für jede mögliche Verbindung Register für die Speicherung der

Akte: PACT16a

ID/Array-ID und Komparatoren zum Vergleich der ID/Array-ID der beiden zu verbindenden Objekte notwendig sind.

Figur 1 zeigt die zwei PAEs (0101, 0102), mit ihren dazugehö-
renden IDs, sowie einen Bus (0103) mit ID. Über die Kompara-
toren (0104, 0105) wird jede PAE/Bus-Verbindung geprüft. Die
Figur soll nur das Grundprinzip verdeutlichen. Sofern alle
Ressourcen (Ein-/Ausgänge der PAEs, Busse) berücksichtigt
werden, steigt die Komplexität und der damit verbundene Hard-
wareaufwand erheblich an.

Ein technisch günstiger zu realisierendes und daher bevorzug-
tes Verfahren wird in den folgenden Abschnitten besprochen:

3. Kontrolle über Vernetzungsstruktur

Abbildung Figur 2 zeigt ein Bus-Segment, das von Konfigurati-
on A und B benötigt wird. Es ist von Konfiguration A belegt.
Konfiguration B kann unabhängig davon die beiden benachbarten
Busstücke bereits belegen. Durch die doppelten Switches wird
ausgeschlossen, dass Daten von Konfiguration B den Datenfluß
von Konfiguration A stören. Ebenso laufen keine Daten von
Konfiguration A nach B. Hierbei muß Konfiguration B wie bis-
her auch darauf vertrauen, dass Konfiguration A korrekt im-
plementiert wurde und der Switch am Ausgang geöffnet ist.

Sobald sich Konfiguration A beendet, wird der freiwerdende Bus
von Konfiguration B belegt, und Konfiguration B beginnt zu
arbeiten.

Mit anderen Worten liegt ein Grundprinzip des Verfahrens
darin, dass jedes Element das an einer Datenübertragung be-
teiligt ist, sich aktiv auf die entsprechende Datenquelle
oder den Datensender aufschaltet.

Akte: PACT16a

Bus zu PAE-Eingang

Abbildung Figur 3 zeigt einen PAE-Eingang (0301), der an die unteren beiden der drei abgebildeten Busse angeschlossen werden soll. Die vertikalen Schalter entsprechen einem einfachen Verbindungsschalter zum Umschalten auf den Bus, der durch die PAE verwaltet werden (0302), die horizontalen Schalter (0303) werden zusätzlich über den Bus konfiguriert um eine korrekte Aufschaltung sicherzustellen.

In Abbildung Figur 3a ist der mittlere Bus noch von einer anderen Konfiguration belegt. Dennoch kann das Objekt mit dem PAE-Eingang vollständig konfiguriert werden. Daten vom mittleren Bus können nicht unbeabsichtigt in das Objekt laufen, da dies durch die Konfiguration des Busses (Schalter 0303) verhindert wird.

In Abbildung Figur 3b hat die alte Konfiguration terminiert und wurde durch die neue Konfiguration ersetzt. Jetzt sind beide Busse verfügbar. Zur Bestimmung, welche Busse tatsächlich aufgeschaltet werden, dienen nur die vertikalen Schalter (0302).

In Abbildung Figur 3c wird schließlich der obere Bus von einer dritten Konfiguration belegt, die ebenfalls den gezeigten PAE-Eingang verwenden möchte. Der Bus wird deshalb bereits so konfiguriert, dass an dieser Stelle Daten entnommen werden können. Dies hat allerdings keinerlei Auswirkungen auf das Objekt, da die PAE Konfiguration an dieser Stelle keine Verbindung vorsieht. Die Verbindung wird also erst hergestellt, wenn die Konfiguration des PAE-Eingangs wechselt.

Akte: PACT16a

Bus--PAE-Ausgang

Dies ist die einzige Verbindung, wo die Verwendung von zwei getrennten Schaltern sinnvoll ist. In den beiden anderen Fällen kann die Funktionalität auch mit einem Schalter realisiert werden, der von zwei Konfig-Bits gesteuert wird. Abbildung Figur 4 zeigt einen PAE-Ausgang, der an die unteren beiden der drei abgebildeten Busse angeschlossen werden soll. Das Objekt wird unabhängig von der Verfügbarkeit der Busse konfiguriert, die jeweils linken Schalter der Abbildung entsprechen der Connection-Mask.

In Abbildung Figur 4a ist der mittlere Bus (0401) noch von einer anderen Konfiguration belegt. Vom Output-Register kann jetzt ein Datenpaket in den Connect geschickt werden. Dieses wird in den angeschlossenen RdyHold (vgl. PACT18) Stufen gespeichert. Das Paket kann durch den geöffneten Schalter des mittleren Busses nicht übertragen und somit auch nicht geACKt werden. Das Objekt kann also keine weiteren Datenpakete übertragen.

In Abbildung Figur 4b wurde der mittlere Bus jetzt umkonfiguriert, so dass auch hier Daten übertragen werden können. Ein evtl. bereits gespeichertes Paket liegt jetzt auf dem Bus, ansonsten funktioniert alles wie früher.

In Abbildung Figur 4c wird der obere Bus (0402) von einer dritten Konfiguration angefordert. Auch hier verhält sich alles wie früher.

Zusammenfassung

Akte: PACT16a

Mit verhältnismäßig geringen Hardwareaufwand lässt sich die Rekonfigurations-Performance erheblich steigern. Es wird dadurch möglich, mehrere vollständige Konfigurationen in die Objekt-FIFOs zu laden.

5

Nach der Rekonfiguration benötigt jedes Objekt lokal nur noch soviel Takte, wie Konfigurationswörter erforderlich sind, bis es wieder konfiguriert ist. Diese Zeit lässt sich durch einen zweiten Registersatz weiter, näherungsweise gegen 0 Takte, drücken.

10

Der zusätzliche Hardwareaufwand beschränkt sich auf ein zusätzliches Konfigurationsbit und AND-Gate pro Switch und pro Anzahl Busse * Anzahl PAE-Eingänge. Die PAE-Ausgänge erfordern je nach Implementierung ein wenig mehr Aufwand, sofern jeweils ein zusätzlicher Schalter notwendig ist.

15

5. Rekonfigurationssteuerung

Die Steuerung der Rekonfiguration wird in der VPU Technologie durch Signale (Reconfig) ausgelöst, die zumeist mit den Datenpaketen und/oder Triggerpaketen über die Bussysteme propagiert werden und anzeigen, dass eine bestimmte Ressource umkonfiguriert werden kann oder soll, ggf. wird gleichzeitig die neue Konfiguration selektiert (vgl. PACT08, PACT13).

25

Sofern ein rekonfigurierbarer Baustein nur partiell umkonfiguriert werden soll, ist es erforderlich Reconfig an bestimmten Stellen entsprechend des Algorithmus zu unterbrechen.

Diese Unterbrechung, die die Weiterleitung von Reconfig verhindert wird als ReconfigBlock bezeichnet.

30

Akte: PACT16a

ReconfigBlock werden für gewöhnlich an den Grenzen einer Konfiguration zur nächsten eingeführt, um diese entsprechend voneinander zu trennen.

5. Zum Versenden von Reconfig bestehen unterschiedliche Strategien, die entsprechend der Anforderung des Algorithmus gewählt werden.

10 Es werden drei mögliche bevorzugte Ausgestaltungen beschrieben, die jeweils einzeln oder kombiniert verwendet werden können und ein unterschiedliches Verhalten aufweisen:

15 a) **ForcedReconfig**: Die einfachste Strategie ist das Versenden des Reconfig über alle Interface eines Objektes. Damit ist sichergestellt, dass alle verbundenen Objekte im PA das Signal erhalten. Zur Einschränkung muß an geeigneten Stellen geblockt werden. Dieses Verfahren, bzw. Signal stellt sicher, dass eine Konfiguration komplett entfernt wird. Das Signal wird im Folgenden mit ForcedReconfig bezeichnet. Das Signal
20 sollte nur verwendet werden, nachdem alle Daten in den betreffenden Objekten verarbeitet und entfernt wurden, da keinerlei Synchronisation mit der Datenverarbeitung stattfindet.

25 b) **SyncReconfig**: Ein Reconfig wird zusammen mit den entsprechenden Daten und/oder Triggern versendet. Das Versenden erfolgt nur zusammen mit aktiven Daten- und/oder Triggerpakete. Das Signal wird bevorzugt zusammen mit dem letzten zu verarbeitenden Daten- und/oder Triggerpakete weitergeleitet und zeigt das Ende der Datenverarbeitung nach diesem Daten-
30 /Triggerpaket an. Benötigt eine PAE mehrere Takte zur Abarbeitung, wird die Weiterleitung von SyncReconfig so lange verzögert, bis das Trigger- und/oder Daten-Paket tatsächlich

Akte: PACT16a

verschickt wird. Somit verhält sich dieses Signal synchron mit der letzten Datenverarbeitung.

- c) **ArrayReset**: Als eine Erweiterung von **ForcedReconfig** kann
5 **ArrayReset** verwendet werden, das nicht geblockt werden kann
und zu einer Rekonfiguration des kompletten Arrays führt.
Dieses Verfahren ist besonders dann sinnvoll, wenn z.B. eine
Applikation terminiert wird oder ein Illegal Opcode (vgl.
PACT19), bzw. Timeout einer Konfiguration auftrat und für ei-
10 ne ordentliche Terminierung der Konfiguration mit anderen
Strategien nicht garantiert werden kann.

5.1 SyncReconfig

- 15 **SyncReconfig** wird immer mit gültigen, aktiven Daten oder
Triggern propagiert.
Probleme treten nunmehr auf, wenn z.B. in einer Verzweigung
das Signal nur in den aktiven Zweig propagiert wird (Figur
5a) oder ein Fork durch fehlende Daten oder Trigger blockiert
20 wird (Figur 5b).
Um dieses Problem zu lösen wird die Semantik von **SyncReconfig**
folgendermassen definiert: Das Signal weist darauf hin, dass
nach Erhalt und vollständiger Verarbeitung der Daten/Trigger
sämtliche Daten-/Triggerquellen (Quellen) und Busse die zum
25 Eingang des Objektes führen, das **SyncReconfig** erhielt, rekon-
figuriert werden. Dazu wird das Signal **ReconfigEcho** einge-
führt. Nach dem Eintreffen von **SyncReconfig** bei einem Ziel-
Objekt wird von diesem **ReconfigEcho** generiert, sobald das
Ziel-Objekt die Daten vollständig verarbeitet hat, die mit
30 dem **SyncReconfig** eintrafen. **ReconfigEcho** wird an sämtliche
Quellen gesendet und führt zur Rekonfiguration der Quellen
und der daten- bzw. triggerübermittelnden Bussysteme.

Akte: PACT16a

Erhält ein Objekt ein ReconfigEcho wird dieses über alle noch aktiven Eingänge an die Quellen des Objektes weiterübertragen.

- Ein/Ausgänge, die bereits ein SyncReconfig erhielten werden durch das Eintreffen passiv, sie führen keinerlei Daten-/Triggertransfers mehr durch. Je nach Ausgestaltung kann ein SyncReconfig nur die Passivschaltung des Eingangs bewirken, bei dem das Signal eintraf oder die Passivschaltung aller Eingänge der PAE.
- 10 Für gewöhnlich trifft eine ReconfigEcho an Ausgängen von PAEs ein. Dies bewirkt die Weiterleitung von ReconfigEcho über die Eingänge der PAE, sofern diese nicht bereits durch ein eingegangenes SyncReconfig passiv geschaltet wurden.
- 15 In einigen Fällen, wie beispielsweise in den Figuren 5a-c, kann ReconfigEcho auch an den Eingängen auftreten. Dies kann je nach Ausgestaltung zur Passivschaltung des Eingangs bei dem das Signal eintraf führen oder in einer bevorzugten Ausführung die Passivschaltung aller Eingänge der PAE auslösen.

20

5.2 Trigger mit Reconfig-Semantik

In einigen Fällen (z.B. Figur 5b) ist eine implizite Propagierung von Reconfig-Signalen (i.b. SyncReconfig, ReconfigEcho) nicht möglich.

- 25 Für die erforderliche explizite Übertragung von beliebigen Reconfig-Signalen kann das Trigger-System nach PACT08 verwendet werden, wobei dazu die Trigger-Semantik entsprechend erweitert wird. Trigger können somit beliebige Status- und Kontrollsignale (wie z.B. Carry, Zero, Overflow, STEP, STOP, GO
- 30 (vgl. PACT08, PACT13, PACT18) übertragen, ebenso wie die impliziten Reconfig-Signale. Weiterhin kann ein Trigger die Semantik SyncReconfig, ReconfigEcho, ForcedReconfig annehmen.

Akte: PACT16a

5.3 Blocken

An jeder Schnittstelle, die ein SyncReconfig versendet, kann eingestellt werden, ob die Versendung stattfinden soll. Das Unterbinden der Propagation führt zu keinen Nebeneffekten. Sofern die Annahme des SyncReconfig an der empfangenden Schnittstelle blockiert wird, schaltet das empfangende Objekt die Schnittstelle, die das SyncReconfig empfang passiv (d.h. die Schnittstelle versendet bzw. empfängt keine Daten mehr), ansonsten reagiert das Objekt nicht auf das Signal, kann jedoch um die Auflösung des übertragenden Bussystemes zu ermöglichen; ReconfigEcho zurücksenden. Desweiteren ist es unabhängig und/oder gemeinsam mit einem ReconfigBlock möglich das ReconfigEcho zu blockieren.

15

5.4 Auswirkung von SyncReconfig und ForcedReconfig auf Bussysteme

Um sicher zu stellen, dass bereits nach dem Durchlauf eines SyncReconfig über einen Bus, keine weiteren Daten übertragen werden, die z.B. von einer nachfolgenden Konfiguration stammen, wird durch SyncReconfig die Weiterleitung von RDY/ACK (vgl. PACT02) über den Bus blockiert. Der Bus selbst wird jedoch nicht abgerissen, um die Rücksendung von ReconfigEcho über das Bussystem zu ermöglichen. Erst mit dem Durchlauf von ReconfigEcho wird der Bus abgebaut.

Andere in der Wirkung äquivalente Methoden können verwendet werden, z.b. könnten Daten- und Triggerverbindungen bereits beim Durchlauf von SyncReconfig abgerissen werden, während die ReconfigEcho-Verbindung erst beim Auftreten von ReconfigEcho abgebaut wird.

12

Akte: PACT16a

Damit ist sichergestellt, dass Daten und Trigger unterschiedlicher nicht zusammengehörender Konfigurationen nicht fälschlicherweise unter den Konfigurationen ausgetauscht werden.

5 Figur 5 zeigt beispielhaft PAEs (0501) mit unterschiedlich konfigurierten Vernetzungen. Die folgenden Übertragungen sind definiert: Daten- und/oder Triggerbusse (0502), SyncReconfig (0503), ReconfigEcho (0504). Weiterhin ist ReconfigBlock (0505) dargestellt. 0506 zeigt an, dass SyncReconfig nicht
10 weitergeleitet wird.

In Figur 5a ist eine Verzeigung abgebildet, wie sie beispielsweise durch ein IF-THEN-ELSE Konstrukt in einem Programm entstehen kann. Nach einer PAE werden die Daten in zwei Pfade verzweigt (0510, 0511) von denen immer nur einer aktiv
15 ist. In dem dargestellten Fall wird ein letztes Datenpaket zusammen mit SyncReconfig übertragen, der Zweig 0510 ist nicht aktiv und leitet daher weder die Daten noch SyncReconfig weiter. Der Zweig 0511 ist aktiv und leitet die Daten und SyncReconfig weiter. Das übertragende Bussystem kann bevorzugt nach der Übertragung bereits inaktiv geschaltet werden
20 und kann lediglich das ReconfigEcho zurückübertragen. Die PAE 0501b erhält das SyncReconfig und sendet es an die PAE 0501c weiter, sie sendet an 0501a das ReconfigEcho zurück, woraufhin 0501a und das Bussystem zwischen 0501a und 0501b rekonfiguriert wird. Die Übertragung zwischen 0501b und 0501c verläuft entsprechend.
25

0501e hat das SyncReconfig von 0501a ebenfalls erhalten, jedoch ist der Zweig nicht aktiv. Daher reagiert 0501e nicht, d.h. 0501e sendet weder das SyncReconfig an 0501f noch das
30 ReconfigEcho an 0501a zurück.

0501c verarbeitet die eingehenden Daten und sendet das SyncReconfig an 0501d weiter. Der Ablauf entspricht zunächst wiederum der Übertragung von 0501a nach 0501b. 0501d gene-

Akte: PACT16a

riert nach der Verarbeitung der Daten ein ReconfigEcho, das aufgrund der Zusammenführung der Zweige auch an 0501f gesendet wird. Obwohl 0501f keine Datenoperation durchführte wird die rekonfiguriert und sendet das ReconfigEcho an 0501e die daraufhin ebenfalls ohne Datenverarbeitung rekonfiguriert wird.

Das von 0501b nach 0501a übertragene ReconfigEcho kann in einer bevorzugten Ausgestaltung auch an 0501e übertragen werden und trifft dort an einem Eingang ein. Dies führt zur Passivschaltung des Einganges und - in einer erweiterten Ausführung, ggf. auch konfigurierbar - zur Passivschaltung aller Eingänge.

Um den Beispielen in Figur 5 einen lokalen Character zu geben wurden die Ein-/Ausgänge der Darstellungen mit einem ReconfigBlock versehen, wodurch die Weiterleitung von SyncReconfig und ReconfigEcho unterbunden wird.

Figur 5b verhält sich weitgehend identisch mit Figur 5a, weshalb auch dieselben Referenzen verwendet werden. Der rechte Pfad ist ebenfalls wieder aktiv, der Linke inaktiv. Wesentlicher Unterschied ist, dass anstatt einer Zusammenführung der Pfade bei 0501d die Pfade nunmehr offen bleiben und beispielsweise direkt an ein peripheres Interface führen. In derartigen Fällen ist es erforderlich zwischen den PAEs (0501i und 0501j) eine explizite Verdrahtung von ReconfigEcho über Triggerleitungen vorzusehen (0507).

In Figur 5c ist die beispielhafte Ausführung einer Schleife dargestellt. Die Schleife läuft über die PAEs 0501m..0501r. Die Übertragungen zwischen den PAEs 0501m..0501r sind entsprechend der vorhergehenden Ausführungen, insbesondere zu den Übertragungen zwischen 0501b und 0501c offensichtlich, da äquivalent.

Akte: PACT16a

Besonders zu beachten ist die Übertragung zwischen 0501r und 0501m. Beim Auftreten von ReconfigEcho an 0501m wird der Bus (0508) zwischen 0501m und 0501r durch die Übertragung von ReconfigEcho rekonfiguriert. ReconfigEcho ist am Ausgang von 0501r geblockt, dadurch wird 0501r nicht rekonfiguriert, der betreffende Ausgang jedoch beim Eintreffen von ReconfigEcho passiv geschaltet, d.h. 0501r leitet keine Ergebnisse mehr auf den Bus. Dadurch kann der Bus von einer beliebigen anderen Konfiguration benutzt werden.

- 5
- 10 Sobald 0501r das ReconfigEcho von 0501q erhält wird 0501r am Ende der Datenverarbeitung rekonfiguriert. Der ReconfigBlock und/oder die Passivschaltung der Busverbindung zu 0501m (0508) verhindern die Weiterleitung in Richtung 0501m. 0501m und 0508 können mittlerweile von einer anderen Konfiguration
- 15 verwendet werden.

6.0 SyncReconfig-II

- Ein weiteres optionales und je nach Anwendung, Einsatzgebiet und/oder Ausgestaltung des Halbleiters oder Systems zu bevorzugendes Verfahren zur Steuerung des SyncReconfig-Protokolls wird nachfolgend beschrieben:
- 20

Das Verfahren ist wie folgt definiert:

- 25 1. SyncReconfig wird prinzipiell über alle verbundenen (Daten- und/oder Trigger-) Busse einer PAE versendet, auch über die, die aktuell (zum jetzigen Takt) keine Daten und/oder Trigger übertragen.
2. Damit eine PAE SyncReconfig entsprechend Abs. 1 weiterleitet, müssen zuvor alle verbundenen Eingänge der PAE das SyncReconfig erhalten haben.
- 30 2a. Rückkopplungen in der Datenstruktur (z.B. Schleifen) erfordern eine Ausnahme des Postulates nach Abs. 2. Die Rück-

Akte: PACT16a

kopplung wird ausgenommen, d.h. es ist ausreichend, wenn alle verbundenen Eingänge einer PAE ausser der oder den schleifenartig zurückgekoppelten das SyncReconfig erhalten haben, damit dieses weitergeleitet wird.

- 5 3. Verarbeitet eine PAE Daten (u.U. auch mehrzyklisch, z.B. Division) wird ein SyncReconfig, wenn dieses entsprechend 2 und 2a an den Eingängen anliegt, zum Zeitpunkt der Fertigberechnung und Weiterleitung der Daten und/oder Trigger an den oder die Empfänger weitergeleitet. Mit anderen Worten überholt kein SyncReconfig die Datenverarbeitung. Nicht datenverarbeitende PAEs leiten SyncReconfig sofort zu den verbundenen Empfängern weiter.

- 10 4. SyncRekonfig wird mit Handshake-Signalen übertragen (z.B. RDY/ACK = Ready/Acknowledge). Eine SyncReconfig absendende
15 PAE geht erst in den Zustand rekonfigurierbar, wenn alle Empfänger den Empfang von SyncReconfig zur Bestätigung mittels eines ACK(nowledge) quittiert haben.

- Bei diesem Verfahren stellt sich grundsätzlich die Frage, was
20 passiert, wenn eine Konfiguration noch nicht vollständig konfiguriert ist, aber bereits wieder rekonfiguriert werden soll. Abseits der Überlegung, ob ein derartiges Verhalten einer Applikation nicht einer besseren Programmierung bedarf, löst sich das Problem wie folgt: Versucht eine PAE das SyncReconfig an eine noch nicht konfigurierte PAE weiterzuleiten, erhält sie so lange kein ACK, bis die PAE konfiguriert ist und das SyncReconfig quittiert. Dadurch entsteht eventuell ein Performance-Verlust, da bis auf die vollständige Konfiguration der zu löschenden Konfiguration gewartet wird, bevor dies gelöscht wird. Andererseits tritt dieser Fall aber
30 äusserst selten und nur unter aussergewöhnlichen Umständen auf.

Akte: PACT16a

Anmelder: Pact XPP Technologies AG
Muthmannstr. 1
80939 München

5

Vertreter: Patentanwalt
Claus Peter Pietruk
Heinrich-Lilienfein-Weg 5
D-76229 Karlsruhe
Vertreter-Nr. 321 605

10

Deutsche Patentanmeldung

Titel: Bussysteme und Rekonfigurationsverfahren

15

Patentansprüche

1. Verfahren zum Ansteuern von Bussystemen dadurch gekennzeichnet, dass
eine Aufschaltung auf einen Bus durch die Genehmigung der
20 Verbindung durch Sender und Empfänger erfolgt.
2. Verfahren zur Rekonfigurationssteuerung von rekonfigurierbaren Elementen, dadurch gekennzeichnet dass
Signale mit den Daten und/oder Trigger übertragen werden die
25 die Rekonfiguration der rekonfigurierbaren Element und/oder Busse steuert.

- 1 / 4 -

Fig. 1

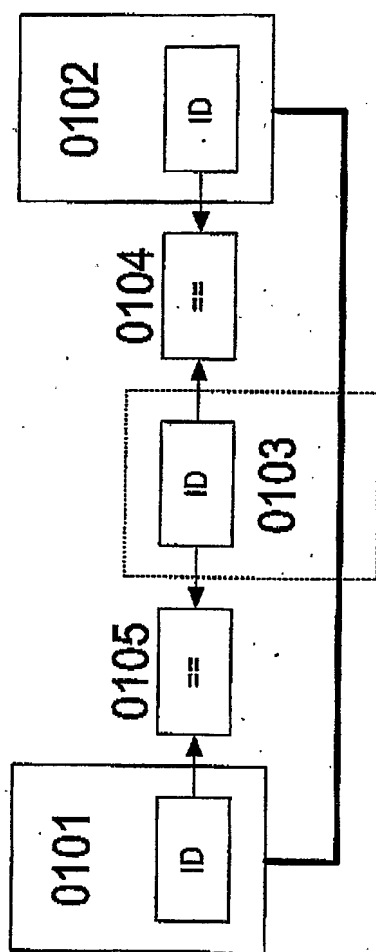
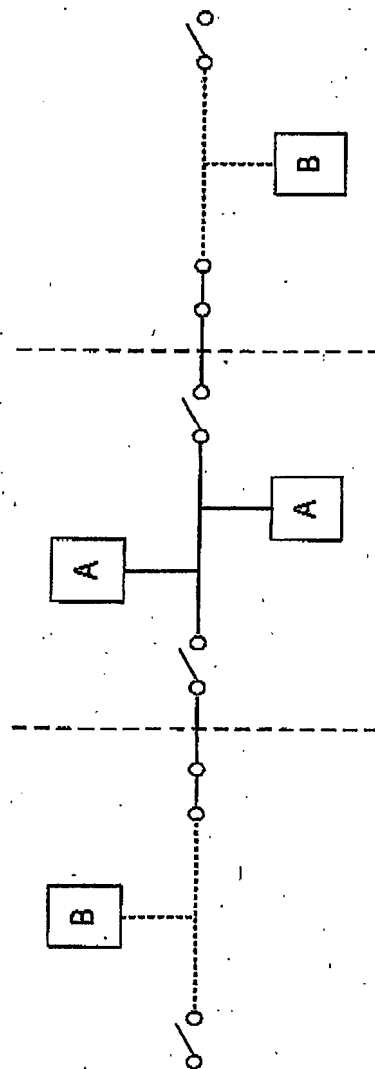


Fig. 2



- 2 / 4 -

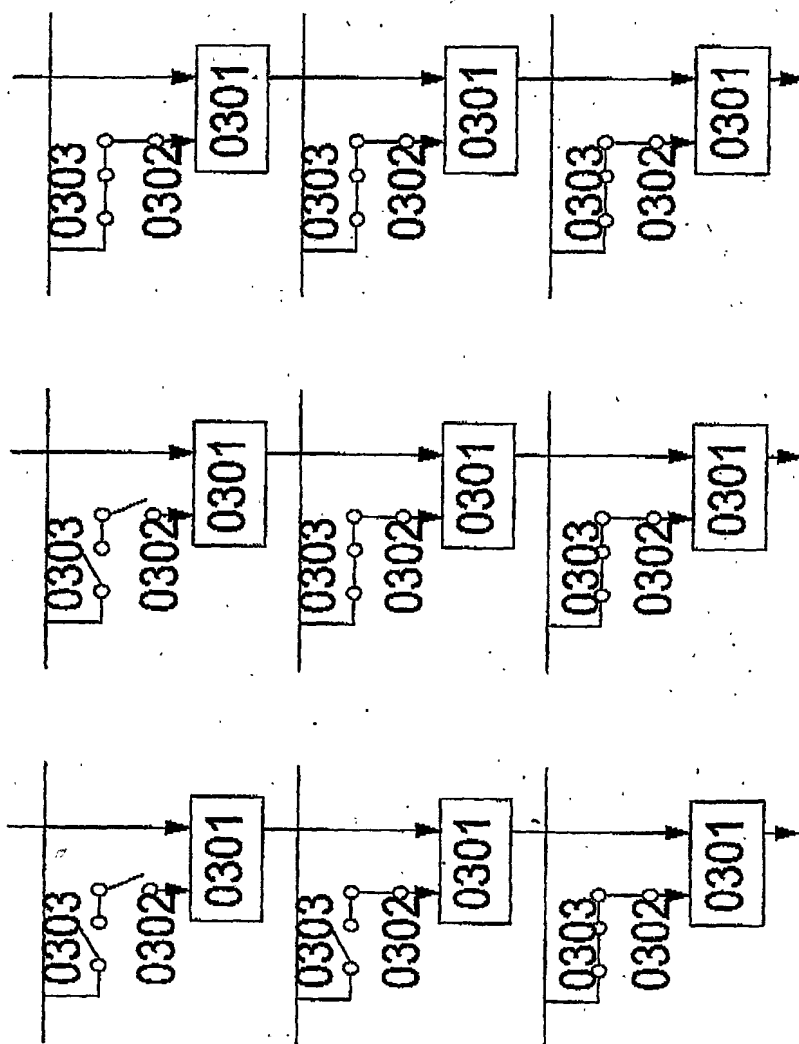


Fig. 3c

Fig. 3b

Fig. 3a

- 3 / 4 -

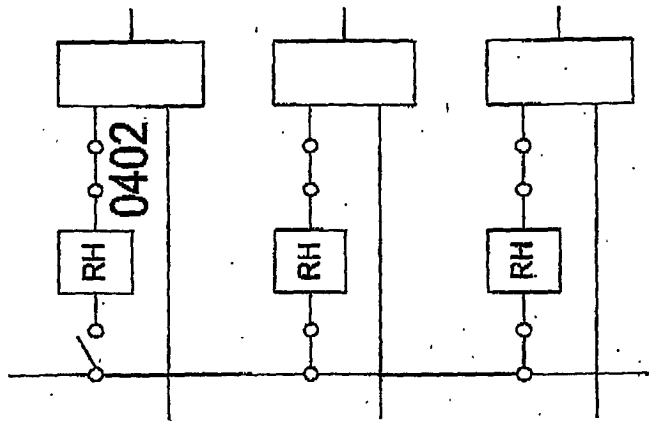


Fig. 4c

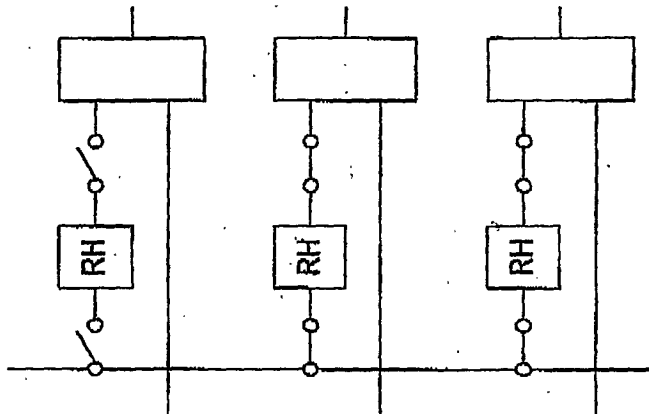


Fig. 4b

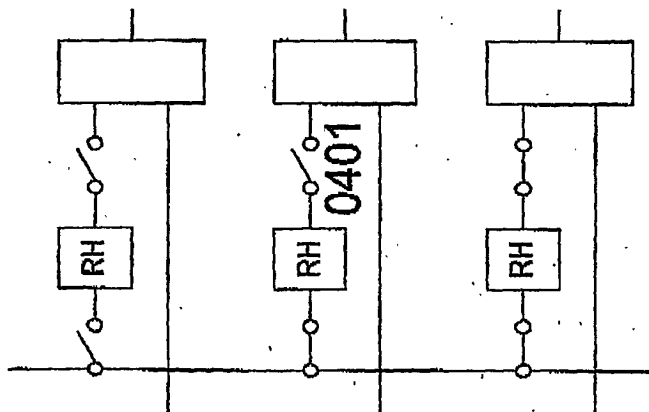


Fig. 4a

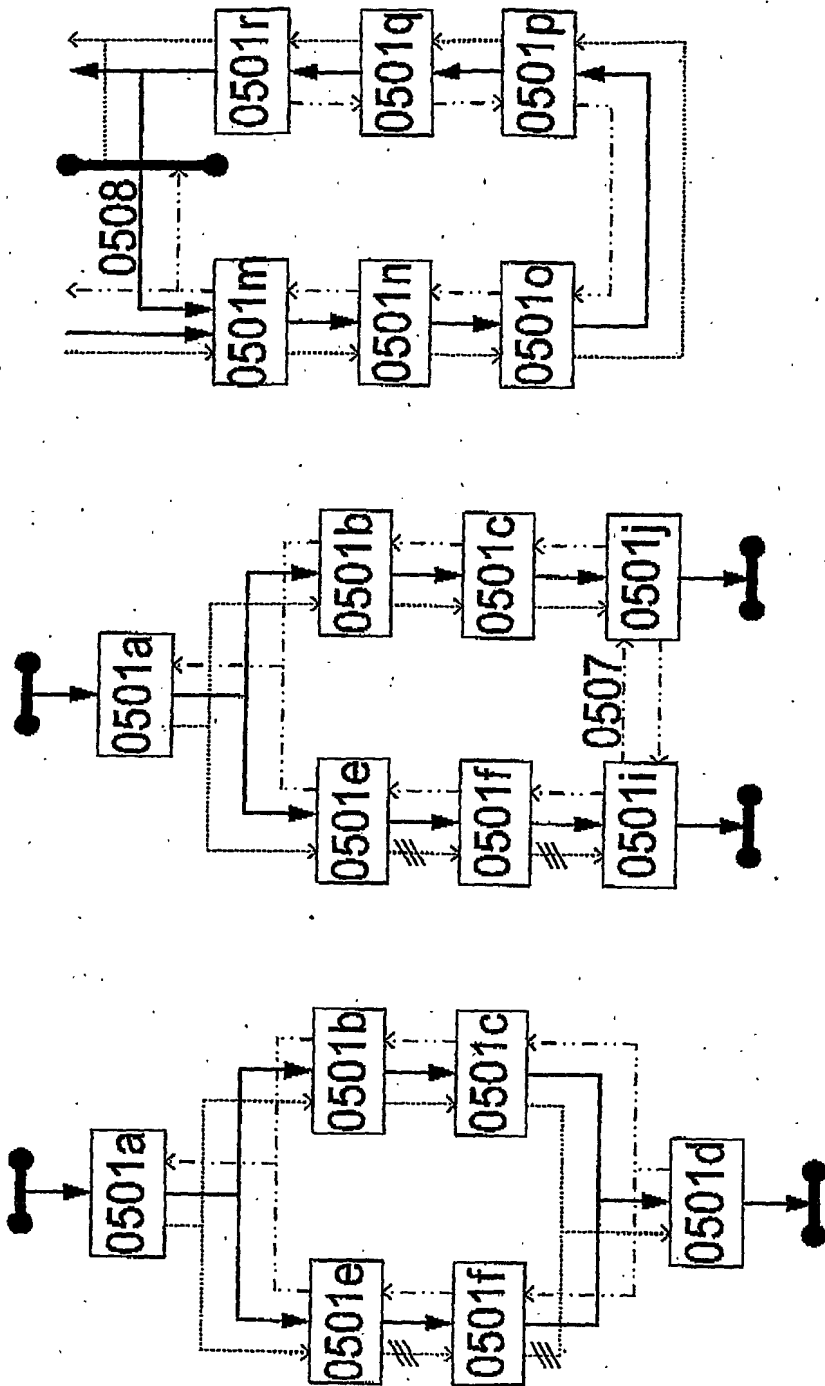
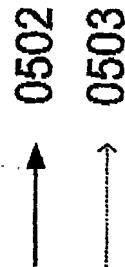


Fig. 5a

Fig. 5b

Fig. 5c



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☐ BLACK BORDERS

☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES

☒ FADED TEXT OR DRAWING

☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING

☐ SKEWED/SLANTED IMAGES

☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS

☐ GRAY SCALE DOCUMENTS

☐ LINES OR MARKS ON ORIGINAL DOCUMENT

☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY

☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.